



Project Report: AstroMart
CS-3773-003 Software Engineering
The University of Texas at San Antonio

November 30th, 2025

Group: 2, Phobost

Price Hiller
Christian Crawford
Joseph Lacap
Daniel Salas
Matthew Ovalle



Contents

Project Description	3
Group Management	4
Roles	4
Contribution	4
Technologies Used	5
Software Architecture	6
Design	7
Landing Page	7
Sign In Page	8
Sign Up Page	8
User Settings/Edit Page	9
Product Editing/Creation Page	10
Example Product Page	11
Product Listings	12
Cart Modal	13
Checkout Page	14
User Orders Page	15
Admin Discount Page	15
Admin Orders	16
All Orders	16
Order Detail	16
Admin User Listing	17
Firestore Data	18
Discounts	18
Products	18
Product Image Storage	19
User Metadata	20
User Order Data	21
Changes from Proposal	22
Lessons Learned	23



Project Description

In our project, we were tasked with creating a fully fledged online shopping site.

While not requiring any specific themes, our team chose to use an outer-space theme to our website, which was displayed through the UI and product choices. Our website firstly was built on the backbone of Google Firebase, which allowed us backend measures to collect and hold user information, order information, discount codes, and product collections. For user authentication, our website was equipped with a detailed sign-in/log-in screen. A user could edit their own details, such as name, their preferred display image, and more. Once the user was logged in, they were greeted by our array of products on sale. The products, accompanied by a description, photo, price, and possible sale information, included common items seen in outer-space, such as “Genuine Moon Rock” and “Meteorite”. Users could then add these products to a shopping cart, which while staying in the current page as a pop-up, allowed them to view the quantity of items selected (with buttons to change or delete the amount), view the subtotal of the cart, and view the overall taxes on the items. Users could then either choose to checkout from the pop-up or the header checkout option and be greeted to a checkout-page detailing their order with an ability to apply any discount codes. Below the cart, the user could finalize the order through payment by inputting their credit card information and finally placing the order. This order, as well as any other orders by the user, could then be monitored through the order screen found by clicking the orders heading on the webpage.

Admin users had a larger amount of freedom on the website, being allowed to dynamically create and change product information, as well as monitor all orders placed on the website, and create and delete discount codes. Product creation and editing relied upon what we've coined the “Phobost API”, which we wrote and deployed on our own infrastructure which converts Markdown to HTML. This permitted users to insert arbitrary Markdown content into the product bodies, significantly easing the creation and editing process of Products.

Overall, the website was fully functional and completely satisfied the rubric.



Group Management

Roles

- **Scrum Master:** Price Hiller
- **Product Owner:** Christian Crawford
- **Front End Developer:** Joseph Lacap
- **Developer:** Daniel Salas
- **Developer:** Matthew Ovalle

Contribution

Price Hiller

Created and built the backend of the website, including the implementation using Google Firebase. Also heavily worked on the frontend of the website through the creation and handling of product implementation, order implementation, user authentication, user editing, multi-page variables, as well as cleaning and reformatting code. Also identified tasks to be done, created issues on Github for task tracking and completion status, as well as communicating deadlines. Beyond that, Price also implemented a complete CI pipeline in Nix to ensure code quality and deployed the backend onto a NixOS server using the primary code base as the deployment source.

Matthew Ovalle

Created along with Joseph LeCapp the website layout design through the use of Figma, including the subsequent helping of implementation of the UI for the design.

Joseph LeCapp

Created along with Matthew Ovalle the website layout design through the use of Figma, including the subsequent helping of implementation of the UI for the design. Also created and helped finalize the shopping cart pop-up, as well as other parts of the website including the search bar.

Christian Crawford

Created and finished the entirety of the checkout page, including the UI and functionality shown in the checkout page as well as helping its integration with other pages.

Daniel Salas

Helped with parts of the shopping cart pop-up, and helped fix multiple issues in the project.

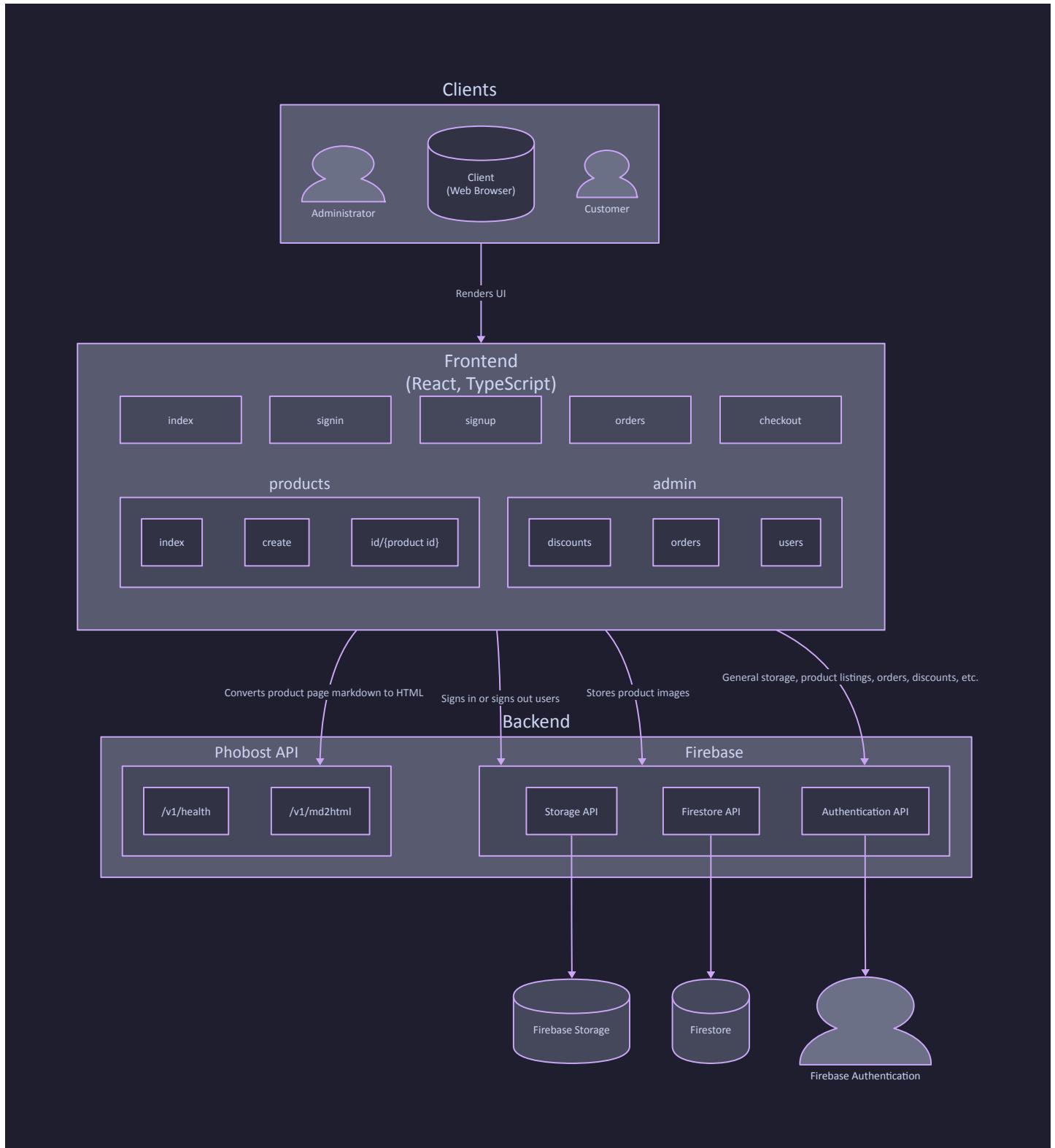


Technologies Used

- Database: Firebase
- Backend: Firebase, Rust + Axum
- Frontend: React, Typescript, HTML, CSS
- CI/CD: Nix
- Deployment: Nix + NixOS, Firebase
- Page Design Outlines: Figma



Software Architecture

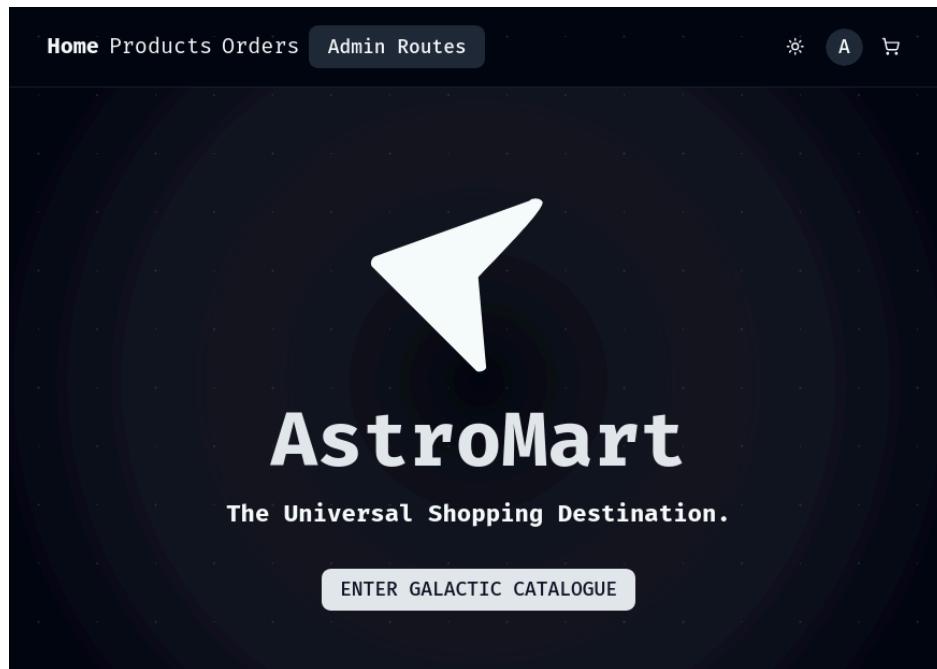




Design

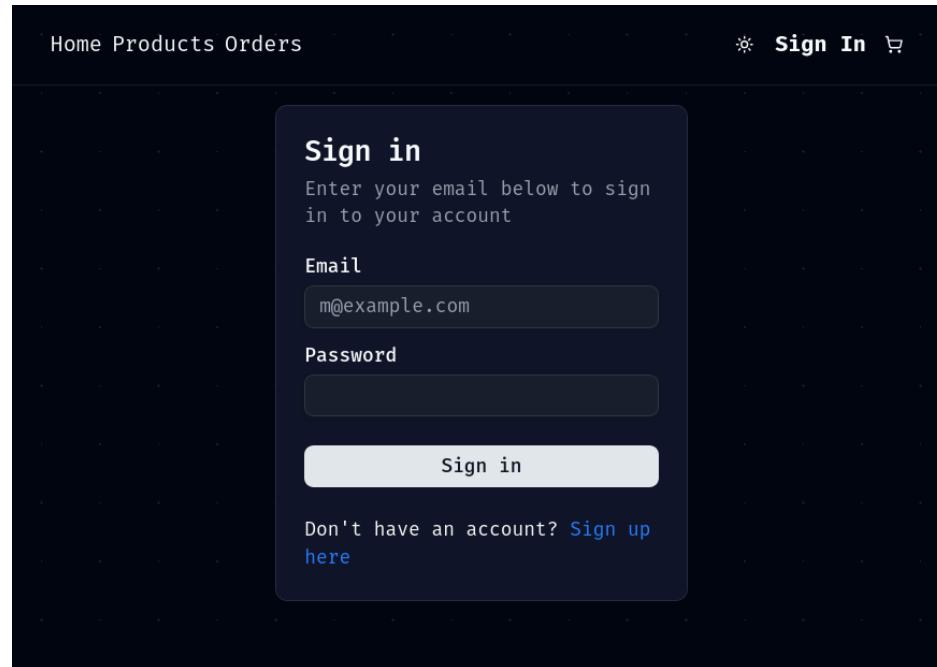
Landing Page

This is the index page (landing page) of the website, allowing access to all others sections within the website. You'll notice the button "Admin Routes" at the top, which only appears if the current user is an admin.



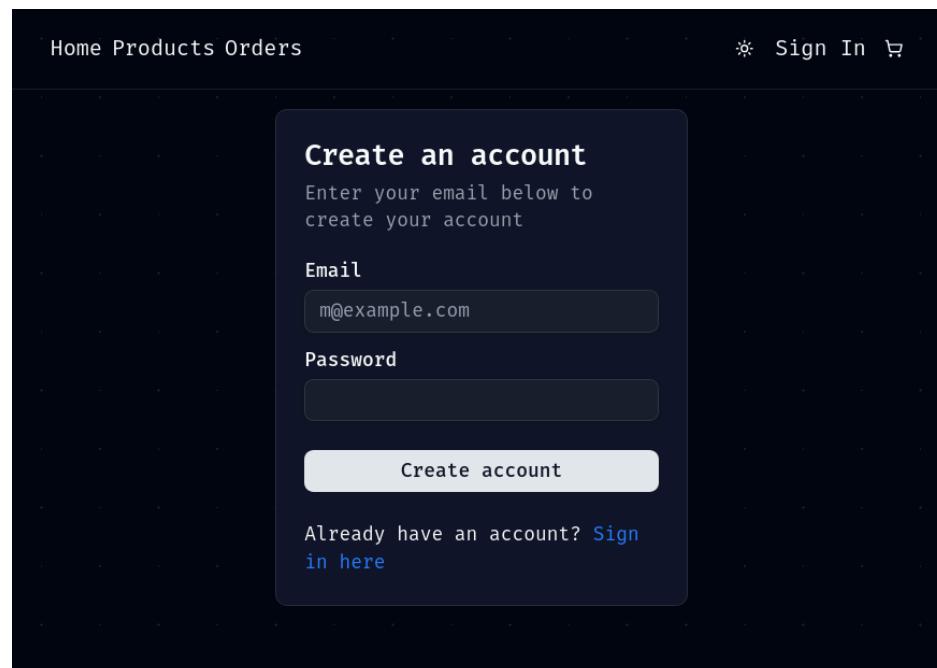


Sign In Page



The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links for "Home", "Products", and "Orders". On the right side of the navigation bar are icons for "Sign In" and a shopping cart. The main content area is a dark box containing a "Sign in" form. The form has a title "Sign in" and a sub-instruction "Enter your email below to sign in to your account". It includes two input fields: "Email" (containing "m@example.com") and "Password" (a redacted field). Below the inputs is a "Sign in" button. At the bottom of the form, there is a link "Don't have an account? Sign up here".

Sign Up Page



The screenshot shows a dark-themed web application interface. At the top, there is a navigation bar with links for "Home", "Products", and "Orders". On the right side of the navigation bar are icons for "Sign In" and a shopping cart. The main content area is a dark box containing a "Create an account" form. The form has a title "Create an account" and a sub-instruction "Enter your email below to create your account". It includes two input fields: "Email" (containing "m@example.com") and "Password" (a redacted field). Below the inputs is a "Create account" button. At the bottom of the form, there is a link "Already have an account? Sign in here".



User Settings/Edit Page

Profile Settings

Manage your account settings and profile preferences.

Display Name

Profile Picture

Or use an image URL:

Note: Uploaded images must be smaller than 900KB

Update Password

Change your account password. You'll need to enter your current password first.

Current Password

New Password

Confirm New Password

Connected Accounts

Manage your connected accounts and authentication methods.

Currently Connected:



Product Editing/Creation Page

Note, this also includes the capability to *preview* the product page before creating/ updating it.

⌚ Toggle Preview

Product Name

Available for Purchase

Set Image

Product Price

\$ 5.49
USD

20

Quantity in Stock

20

Sale Percentage

21

Short Product Description

Snacks that taste like chicken ... if chicken were neon green and crunchy.

No more than 120 characters | (73/120)

Content Body

The content displayed on the detailed product page.

```
# Basically Cosmic Trailmix

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent nec sollicitudin dolor. Suspendisse convallis est erat, ac ornare velit euismod ac. Pellentesque eu felis ac nunc tempus aliquet at pretium magna. Curabitur rutrum dolor in pulvinar porta. Maecenas in ligula in nibh semper mattis. Etiam porta maximus tincidunt. Phasellus ac elit non purus fringilla ornare tempor sit amet risus. Curabitur tempor posuere libero nec maximus.

Fusce nec ipsum lacinia, venenatis tellus id, iaculis diam. Donec vitae nulla ut libero sagittis auctor id nec magna. Cras egestas dictum massa, ac placerat neque congue eu. Phasellus magna nisl, convallis vel ligula ac, dapibus tempor lorem. Proin semper dictum egestas. Suspendisse et tortor efficitur, scelerisque enim eget, vulputate sapien. Cras eleifend nisl ac viverra sodales. Aliquam vitae metus semper, vehicula eros eu, mollis enim.
```



Example Product Page

Note that the **Edit** button appears *only* when the current user is an admin.

[Back to shop](#) [Edit](#)

Coende Crunch Alden

\$5.49 \$4.34 21% off!

Snacks that taste like chicken... if chicken were neon green and crunchy.

Add to Cart Buy now



Basically Cosmic Trailmix

Lore ipsum dolor sit amet, consectetur adipiscing elit. Praesent nec sollicitudin dolor. Suspendisse convallis est erat, ac ornare velit euismod ac. Pellentesque eu felis ac nunc.



Product Listings

You can search the products by both **Title** and **Description** in the search box, as well as sort by Price and Availability.

Products

Search products ...

↑↓ Price ↑↓ Availability

+ New Product

Coende Crunch Alden

~~\$5.49~~ \$4.34 21% off!

Snacks that taste like chicken... if chicken were neon green and crunchy.

In Stock: 20

Add to Cart Buy now Details

Bottle of Stardust

~~\$29.99~~ \$23.69 21% off!

For sprinkling on your cereal or wishing upon. Contains glitter.

In Stock: 8

Add to Cart Buy now Details

Pluto

\$12.00

The entire dwarf planet, seriously.

In Stock: 7

Add to Cart Buy now Details



Cart Modal

The below is a user's cart, prior to them hitting checkout.

They can change the quantity of items in the cart with the plus and minus buttons as well as completely remove items by hitting the trash icon or clearing the entire cart by hitting the trash icon under the "Total" entry.

Coende Crunch	-	3	+	
Alden				
Bottle of Stardust	-	1	+	
Pluto	-	1	+	
<hr/>				
Subtotal:			\$48.70	
Tax:			\$4.02	
Total:			\$52.72	
<hr/>				
Checkout				



Checkout Page

The checkout also permits the usage of a discount code to apply a singular percentage reduction discount to the current subtotal.

Checkout

This is your cart & checkout page. Review your order before completing the purchase.

Order summary

Filter products ...

Product ↑↓	Product	Price ↑↓
3	Coende Crunch Alden	\$16.47 \$13.01
1	Bottle of Stardust	\$29.99 \$23.69
1	Pluto	\$12.00

Previous Next

Discount Code Apply

Subtotal:	\$48.70
Tax:	\$4.02
Order Total:	\$52.72

Payment Details

Card Number 0000 0000 0000 0000

Name on Card Cardholder Name

Expiry Date (MM/YY) MM/YY CVC 123

Pay \$52.72 & Place Order



User Orders Page

This page displays only the orders for the current user along with the order status.

Filter orders by id ...		Filter orders by customer ...		Filter by Status	
Order Date ↑	Order ID	Customer ↑	Total ↑	# Products ↑	Status ↑
⌚ 12/1/2025, 4:13:52 AM	zm0wFnJdQXdWmlQJtdx0	a@a.com	\$52.72	3	Pending
1 orders					

Admin Discount Page

This page allows admins to create and delete discounts for use at checkout by users.

Filter codes ...		Create Discount
Code	Percentage ↑	
another	30%	
code	5%	
example	10%	
last	45%	
4 discounts available		



Admin Orders

All Orders

This displays *all* orders site-wide for all users for admin to view.

Filter orders by id ...		Filter orders by customer ...		Filter by Status	
Order Date ↑	Order ID	Customer ↑	Total ↑	# Products ↑	Status ↑
🕒 12/1/2025, 4:16:37 AM	nQ2Tr3PmJ4NDyP0s4RT0	a@a.com	\$111.15	2	Pending
🕒 12/1/2025, 4:13:52 AM	zm0wFnJdQXdWmlQJtdx0	a@a.com	\$52.72	3	Pending
2 orders					

Order Detail

This displays the specific details of an order.

Order: nQ2Tr3PmJ4NDyP0s4RT0				X
Pending				
Product	Cost per Unit	Quantity	Cost	
Bottle of Stardust	\$29.99	3	\$71.08	
Genuine Moon Rock	\$20.00	2	\$31.60	
Subtotal			\$102.68	
Tax			\$8.47	
Total			\$111.15	
Ordered: 12/1/2025, 4:16:37 AM				



Admin User Listing

This page allows admins to view all users on the site and change their admin status.

Admin User Listing			
Email ↑	User ID	Registration Date ↑	Admin
price@pricehiller.com	AFBclrAubF6AktV1JJD2KypepPcE	12/1/2025	<input checked="" type="checkbox"/>
a@a.com	ByKEBvndjavEUtXQ5yGtupD2B5Kj	11/25/2025	<input checked="" type="checkbox"/>



Firebase Data

Discounts

Root	discounts	another
+ Start collection	+ Add document	+ Start collection
discounts	another	+ Add field
products	code	id: "another" (string) edit trash
users	example	percentage: 30 (number) edit trash
	last	

Products

Root	products	6UdS1YAabvWTUyomRpaM
+ Start collection	+ Add document	+ Start collection
discounts	6UdS1YAabvWTUyomR...	+ Add field
products	GJxosRCw251lQuEMb...	- body (map) + trash
users	I1KGYLteJS3d9V8Zv...	html: "<h1><a href="#ba... (string) edit trash
	KtaUpylrlDydNr0MI...	markdown: "# Basically ... (string) edit trash
		description: "Snacks that ... (string) edit trash
		isAvailable: true (boolean) edit trash
		name: "Coende Crunch Alden" (string) edit trash
		price: 5.49 (number) edit trash
		primaryImageUrl: "http://1... (string) edit trash
		quantityInStock: 17 (number) edit trash
		salePercentage: 21 (number) edit trash



Product Image Storage

gs://swe-cs3773-g2-da56b.firebaseiostorage.app > products				
<input type="checkbox"/>	Name	Size	Type	Last Modified
<input type="checkbox"/>	6UdS1YAabvWTUyomRpaM/	—	Folder	—
<input type="checkbox"/>	GJxosRCw251lQuEMbJFj/	—	Folder	—
<input type="checkbox"/>	GvKmOvbPUrQmxVo0g1K/	—	Folder	—
<input type="checkbox"/>	IIKGYLteJS3d9V8Zvt2R/	—	Folder	—
<input type="checkbox"/>	6UdS1YAabvWTUyomRpaM	312.69 kB	image/jpeg	12/1/2025
<input type="checkbox"/>	GJxosRCw251lQuEMbJFj	143.38 kB	image/jpeg	12/1/2025
<input type="checkbox"/>	GvKmOvbPUrQmxVo0g1K	359.82 kB	image/png	11/25/2025
<input type="checkbox"/>	IIKGYLteJS3d9V8Zvt2R	300.92 kB	image/jpeg	12/1/2025

Note that the seemingly random names there are actually the product IDs.



User Metadata

Root	users	AFBclrAubF6AktV1JJD2KypepPcE	⋮
+ Start collection	+ Add document	+ Start collection	⋮
discounts	AFBclrAubF6AktV1J...	orders	
products	ByKEBvndjavEutXQ5...		
users		+ Add field	
		- cart (map) + ⚡	
		items: [] (array) + ⚡	
		updatedAt: Mon Dec ... (timestamp) ⚡ ⚡	
		userId: "AFBclrAubF6Akt..." (string) ⚡ ⚡	
		- settings (map) + ⚡	
		base64Image: "data:imag..." (string) ⚡ ⚡	
		displayName: "" (string) ⚡ ⚡	
		email: "price@pricehill..." (string) ⚡ ⚡	
		photoURL: "https://ui-a..." (string) ⚡ ⚡	



User Order Data

The screenshot shows the Firebase Firestore interface for managing user order data. The document structure is as follows:

```

  users > AFBclrAubF6AktV1JJD2KypepPcE > orders > EYNUEaNdqX0qbV70e5j8
    - discount: null
    - products:
      - 0:
        - id: "GJxosRCw251lQuEM..." (string)
        - name: "Bottle of Star..." (string)
        - price: 29.99 (number)
        - quantityOrdered: 2 (number)
        - salePercentage: 21 (number)
        - status: "Pending" (string)
        - subtotal: 47.3842 (number)
        - tax: 3.9091965 (number)
        - timestamp: Mon Dec 01 ... (timestamp)
        - total: 51.2933965 (number)
        - userId: "AFBclrAubF6AktV1J..." (string)
    - cart:
      - items: (array)
        - updatedAt: (timestamp)
      - userId: (string)
    - settings:
      - base64Image: (string)
      - displayName: (string)
      - email: (string)

```

Note that orders are stored under each user element and then a grouped query is made to pull all of them for the admin orders listing.



Changes from Proposal

In our initial proposal we intended to build a full backend in Python for most of the data handling. One of our members joined late and gave an alternative proposition of using Firebase due to their prior experience with the platform. This eventually resulted in the complete removal of any Python backend and SQL data storage, instead opting to use Firebase's Firestore, Storage, and Authentication services. This decision largely wiped out writing most backend code as Firebase handles almost all significant backend services for us.

If you compare our project proposal's architecture diagram to the architecture diagram provided in this report, you'll notice a massive difference in where complexity is located. Firebase moved much of the complexity towards the frontend as we could focus more of our effort towards it due less work being required on the backend. The only significant backend component that survived the switch to Firebase was the "Product Page Service", which is the "Phobost API" in this report's software architecture diagram.



Lessons Learned

We gained significant experience using Firebase, Typescript, and React, skills few of us previously possessed. We also learned how to integrate Github CI/CD flows for building and handling deployments of applications. For instance, we used Nix to build the code assets and then Nix & NixOS to deploy some of them onto a live server for later usage.

We also learned how to coordinate work using Github issues, such that team members were not duplicating work-in-progress. On top of this, we learned how to work with Git, to create branches, add changes, push branches to origin, opening pull requests to main, reviewing pull requests, and merging in said pull requests to main. This closely aligns with real-world practices in managing software development, especially as applied to Git repositories.

A significant lesson though, that comes to mind, is that Firebase can be problematic in some cases as you're entirely bound to (and at the mercy of) Google's services for your data storage, authentication, and other primary backend needs. At the time of writing, Price's account has a hold on it from Google meaning he cannot deploy the website entirely in Firebase and is instead dependent on the local emulators to "deploy" the project. Google raised a potential fraud issue with the credit card on file and now an automated remediation process is occurring on Google's side, until that is done we cannot fully deploy the application. Firebase brings with it great powers, largely eliminating many core backend service requirements, but if you have any issues with billing, Google's policies, or other services attached to Google, you end up having to wait on Google to resolve the problem. You cannot fully, independently, host the entire service infrastructure for the application, creating a strong dependency issue if you have any issues with Google or Firebase services. In the cases where you need the flexibility of managing the infrastructure directly, or being able to modify it, it may be a better idea to accept the up-front cost of writing your own core backend yourself to avoid these potential issues.